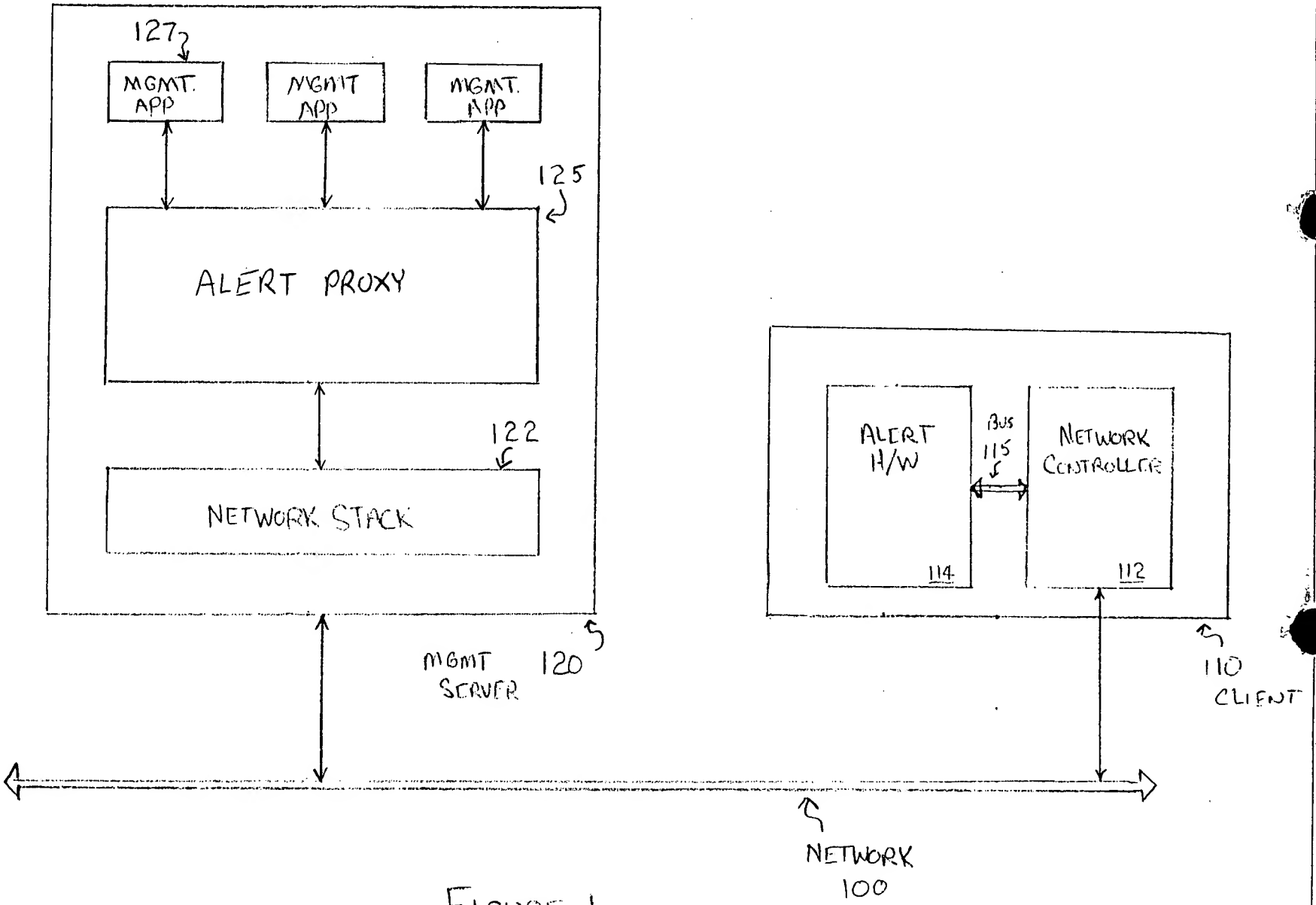

127
MGMT. APP
MGMT APP
MGMT. APP
125
ALERT PROXY
122
NETWORK STACK
MGMT SERVER
120
ALERT H/W
114
BUS 115
NETWORK CONTROLLER
112
110 CLIENT
NETWORK 100

FIGURE 1

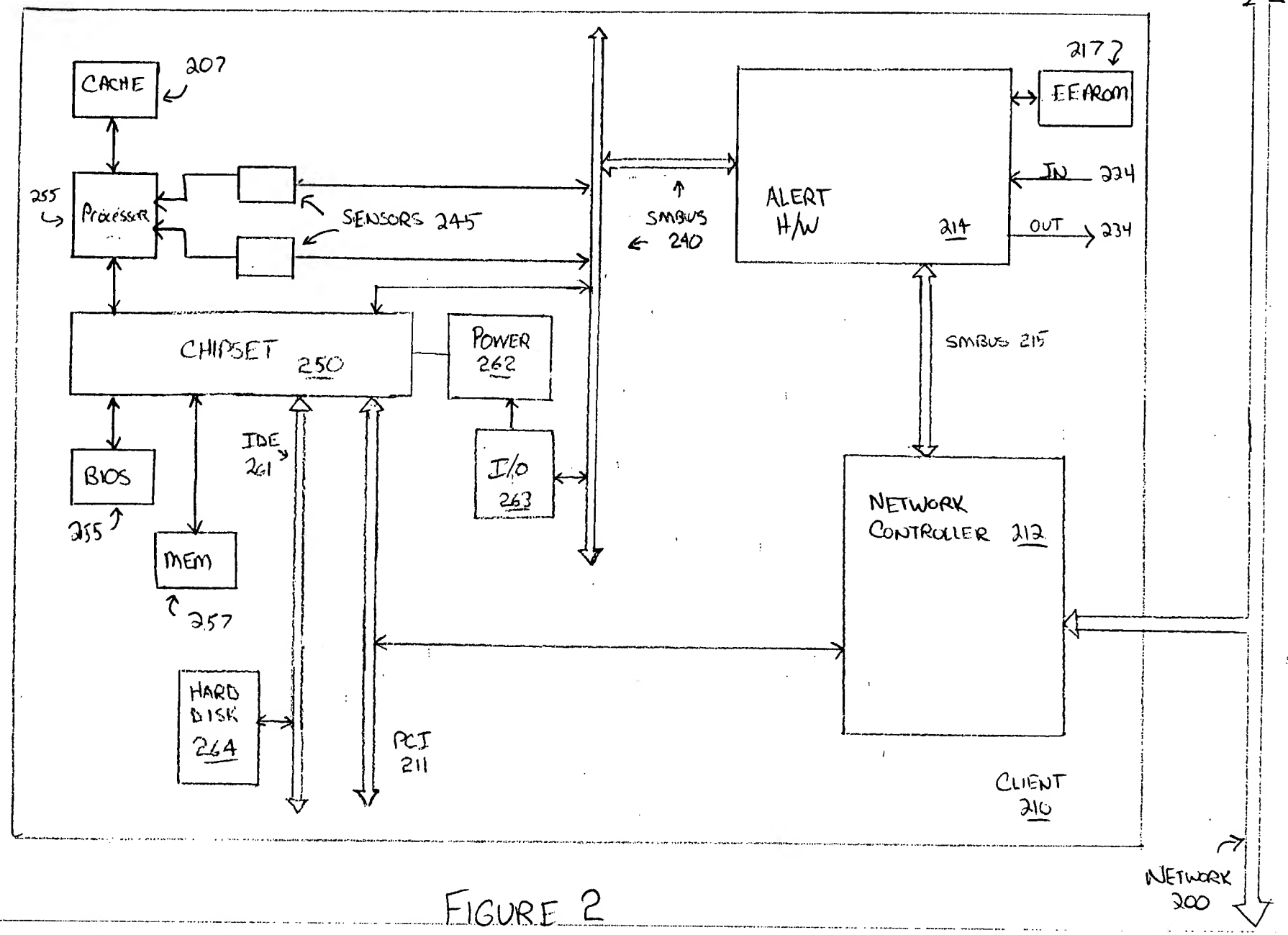

CACHE
207
255
Processor
SENSORS 245
SMBUS 240
ALERT H/W
214
217
EEPROM
IN 224
OUT 234
CHIPSET 250
POWER 262
SMBUS 215
IDE 261
BIOS
255
I/O 263
MEM
257
NETWORK CONTROLLER 212
HARD DISK 264
PCI 211
CLIENT 210
NETWORK 200

FIGURE 2

Figure 3A

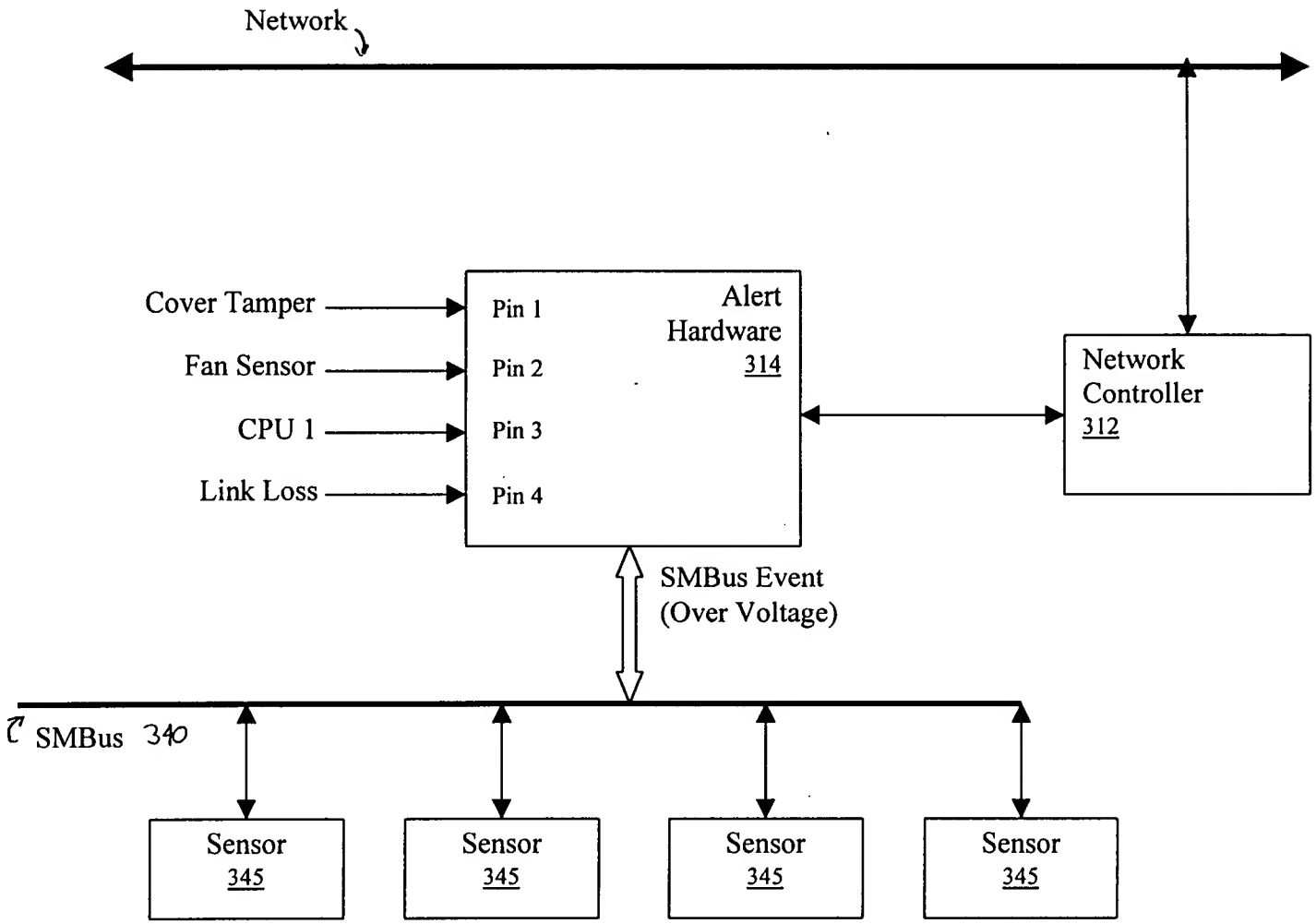

Network
Cover Tamper
Fan Sensor
CPU 1
Link Loss
Pin 1
Pin 2
Pin 3
Pin 4
Alert Hardware
314
Network Controller
312
SMBus Event
(Over Voltage)
SMBus 340
Sensor
345
Sensor
345
Sensor
345
Sensor
345

Figure 3B

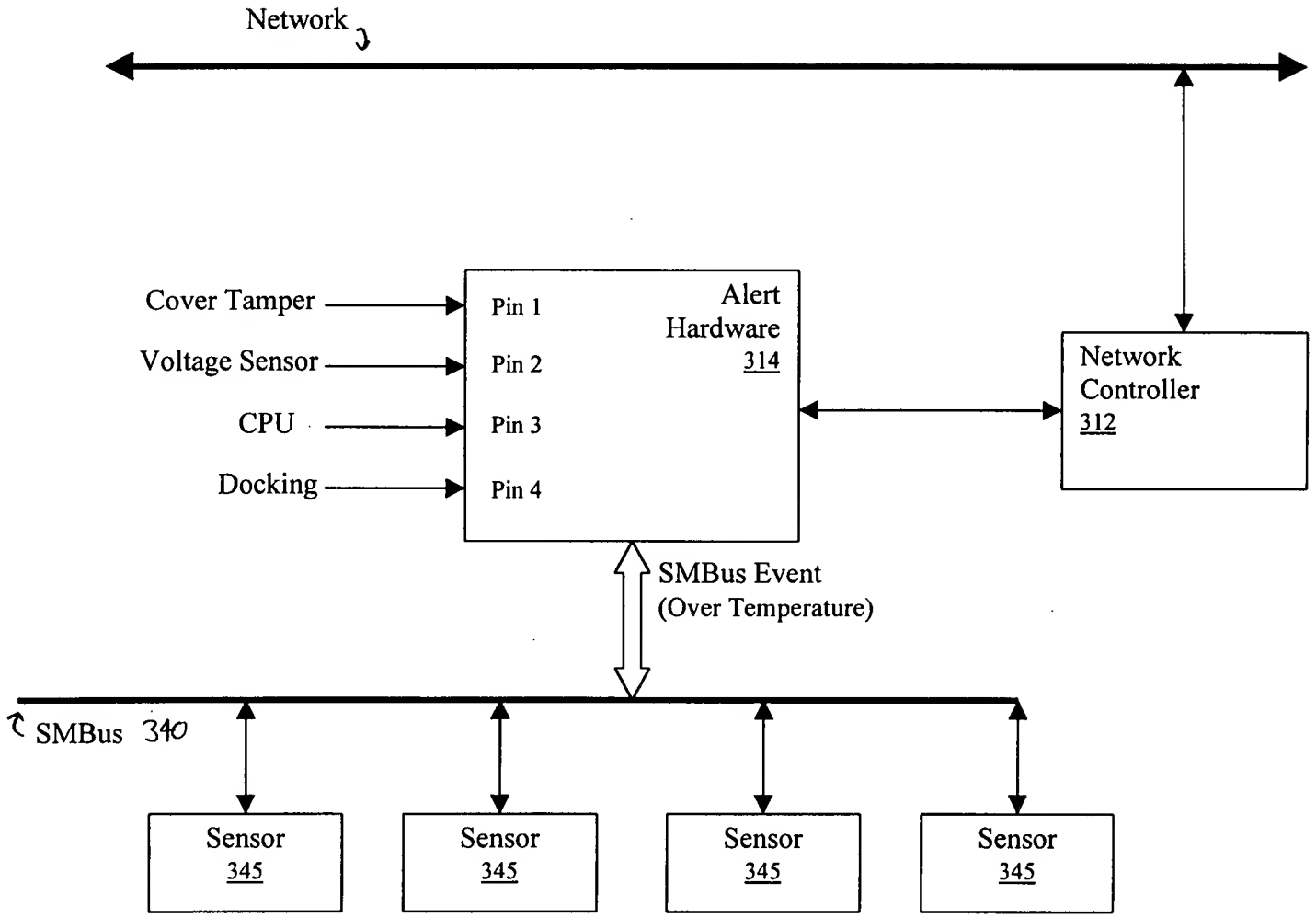

Network
Cover Tamper
Voltage Sensor
CPU
Docking
Pin 1
Pin 2
Pin 3
Pin 4
Alert Hardware 314
Network Controller 312
SMBus Event (Over Temperature)
SMBus 340
Sensor 345
Sensor 345
Sensor 345
Sensor 345

Figure 4A

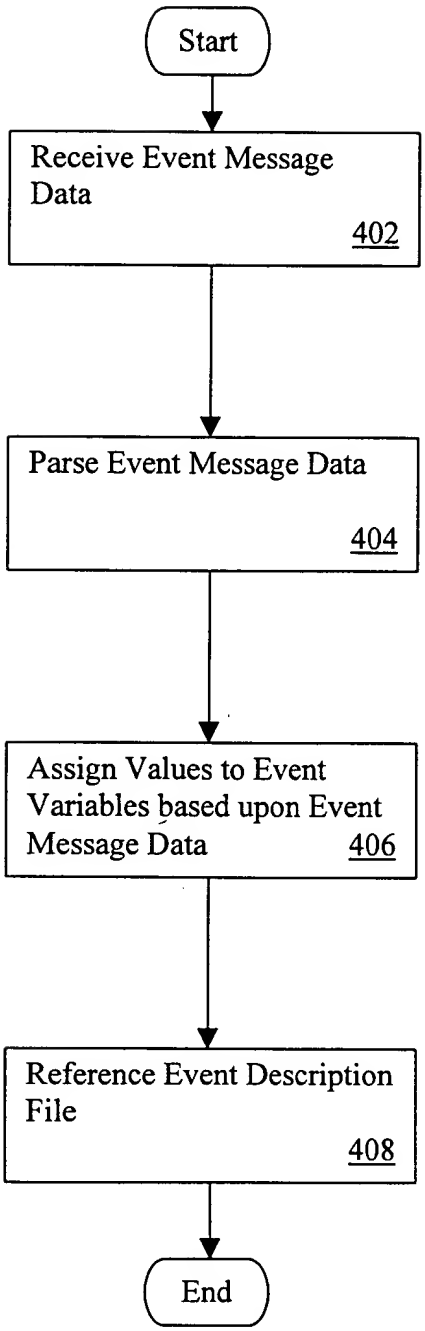

Start
Receive Event Message Data
402
Parse Event Message Data
404
Assign Values to Event Variables based upon Event Message Data
406
Reference Event Description File
408
End

Figure 4B

422	MESSAGE TYPE	Identifies Specific Alert Hardware
424	SYSTEM ID	Identifies Specific Platform Type
426	UUID	Identifies A Client Entity
428	EVENT TYPE	Identifies Alert Hardware Event Type
430	MESSAGE	Contains A Software Message Used By "Software" Events
432	EVENT EXTENSION	Contains Descriptor Data Used By "Compound" Events
434	EVENT DATA	Contains Additional Event Information

Figure 5A

505 → [LANGUAGE_MAP]
esm = esp

510 → [SYSTEM ID]
511 → XXX = CompanyXYZ
512 → YYY = CompanyABC

520 → [EVENT_TYPE]
0 = SIMPLE
521 → 1 = SIMPLE
522 → 8 = SOFTWARE
523 → 90 = COMPOUND

530 → [EVENT_MAP]
531 → CompanyXYZ = 1,2; 2,2; 84,9; 90,7

540 → [EVENT_LIST]
0 = 13000;14014
541 → 1 = 13001;14015
2 = 13002;14016
3 = 13003;14017

550 → [COMPANYXYZ_90]
551 → 7 = 8
552 → 9=10
[RESOURCE_MAP]
CompanyXYZ = XYZ
CompanyABC = ABC

565 → [COMPANYXYZ]
0106=2712

FIGURE 5B

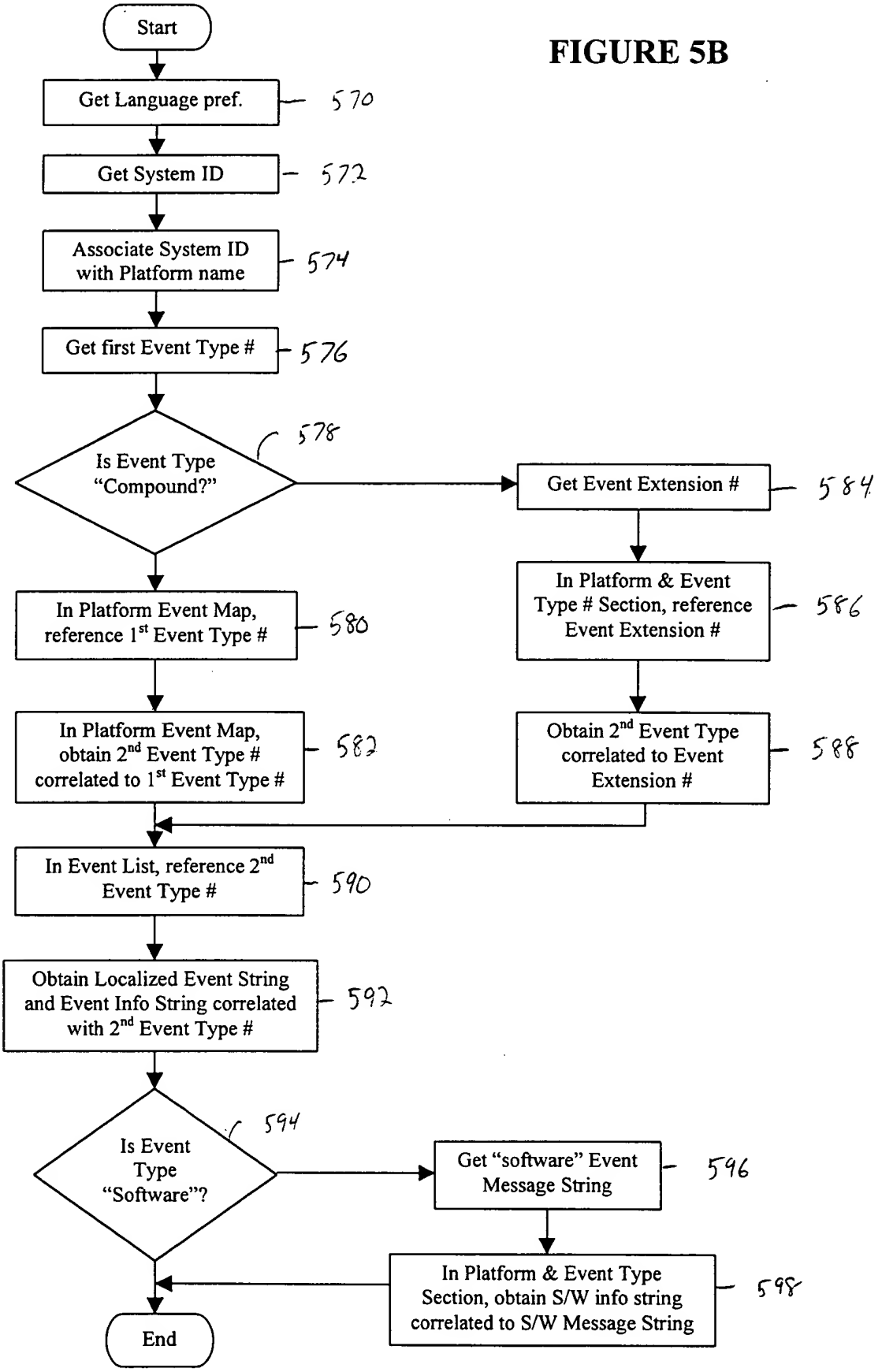

Start
Get Language pref.
570
Get System ID
572
Associate System ID with Platform name
574
Get first Event Type #
576
Is Event Type "Compound?"
578
Get Event Extension #
584
In Platform & Event Type # Section, reference Event Extension #
586
In Platform Event Map, reference 1st Event Type #
580
In Platform Event Map, obtain 2nd Event Type # correlated to 1st Event Type #
582
Obtain 2nd Event Type correlated to Event Extension #
588
In Event List, reference 2nd Event Type #
590
Obtain Localized Event String and Event Info String correlated with 2nd Event Type #
592
Is Event Type "Software"?
594
Get "software" Event Message String
596
In Platform & Event Type Section, obtain S/W info string correlated to S/W Message String
598
End

Figure 6A

```
610 → [FUNCTION_LIST]
      0 = 20000;21015
      1 = 20001;21016
616 → 2 = 20002;21017

620 → [FUNCTION_MAP]
621 → COMPANYXYZ = 1,1;2,2;3,3;4,4;5,5;9,6

630 → [SUBFUNCTIONS_2]
632 → 0 = 16001;16002
634 → 1 = 16003;16004

      [COMPANYXYZ]
      DeviceType = 0
      StatusPolarity = 0xD9
      StatusPowerupMask = 0x1B
      StatusPowerdownMask = 0x19

      [CompanyXYZ]
652 → 101   I:300401
      110   W:300402
      114   C:300403

      [CompanyABC]
662 → 101   300402;300800;300801
      110   300405
```

Figure 6B

672 →	300401	A failure occurred during testing of the system board
	300402	A memory parity failure occurred
	300403	Adapter ROM failure
	300800	Memory may be defective
	300801	Replace memory

Figure 7

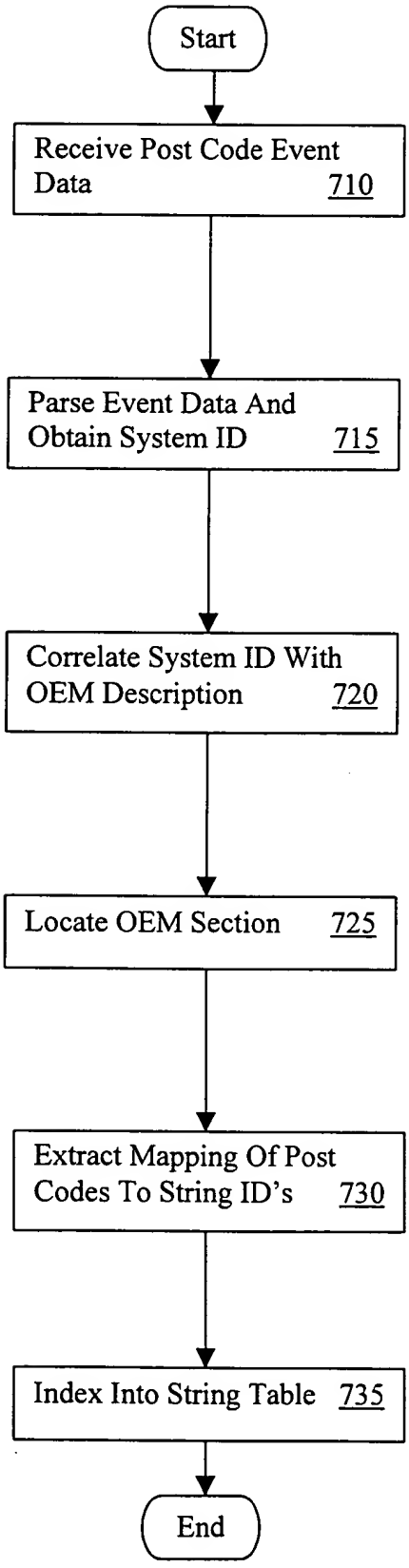

Start
Receive Post Code Event Data 710
Parse Event Data And Obtain System ID 715
Correlate System ID With OEM Description 720
Locate OEM Section 725
Extract Mapping Of Post Codes To String ID's 730
Index Into String Table 735
End

FIGURE 9

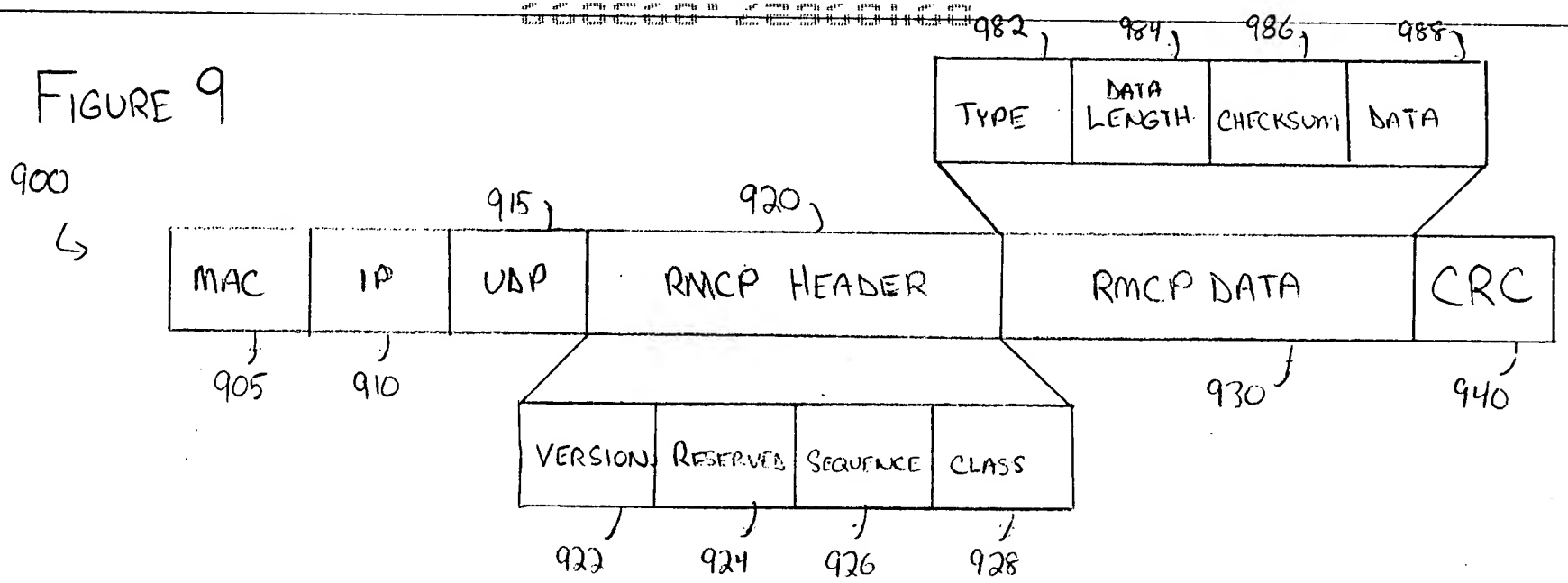


FIGURE 8

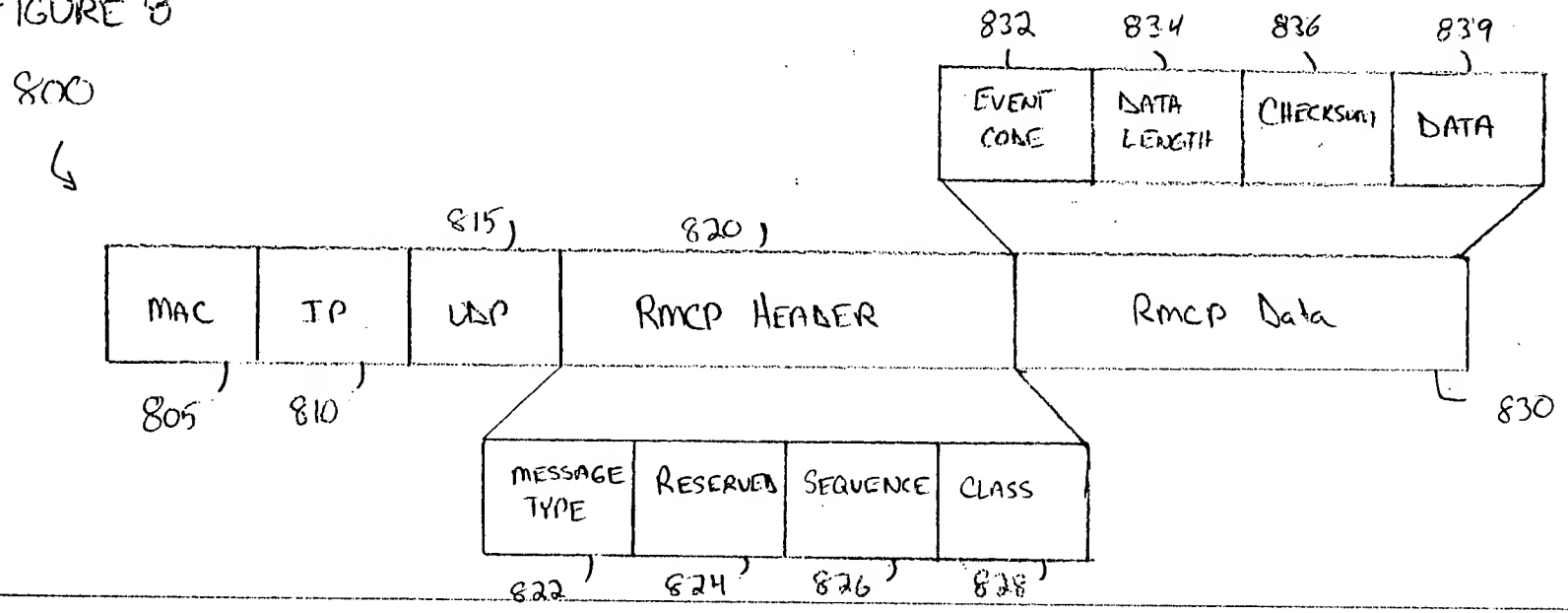


FIGURE 10

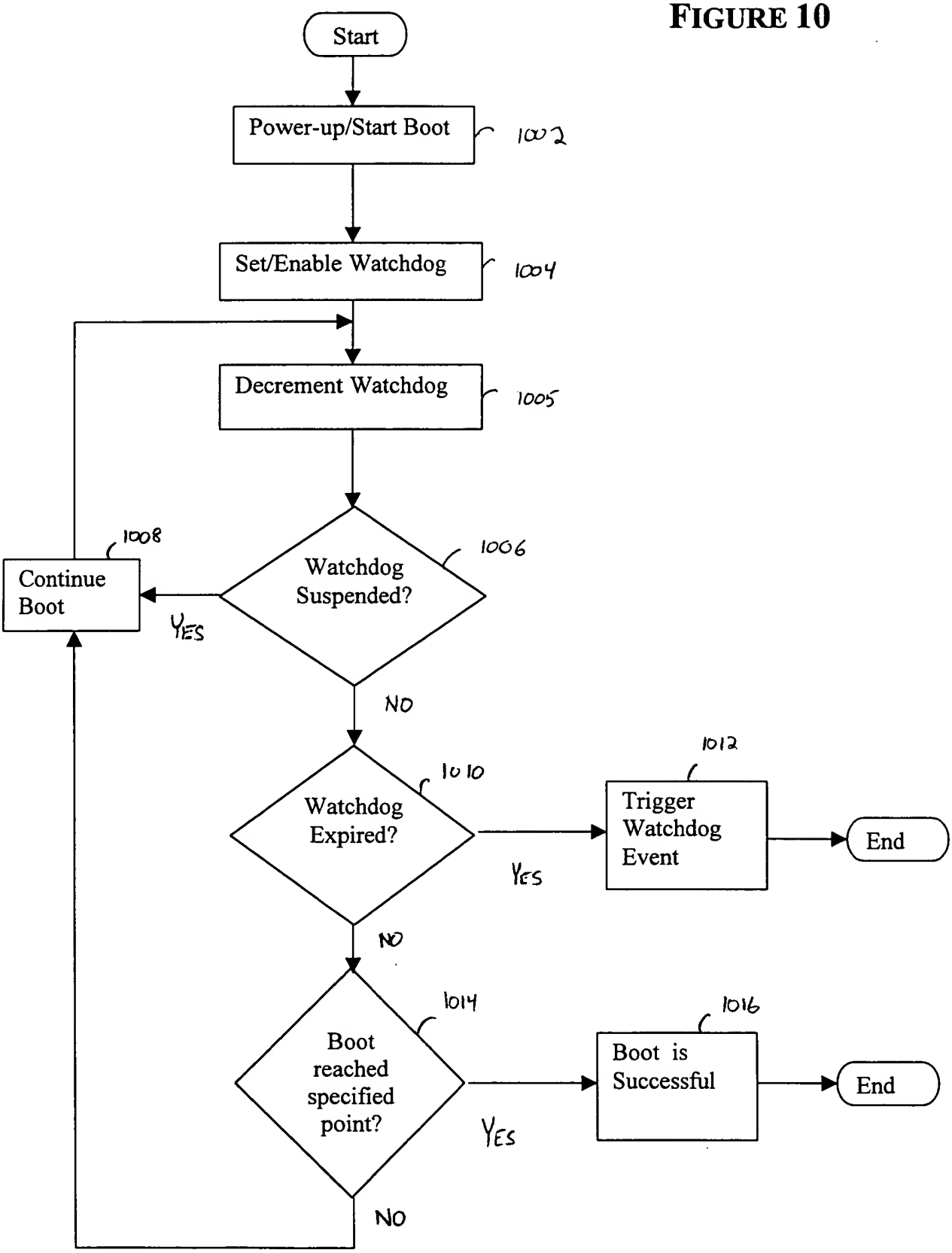

Start
Power-up/Start Boot
1002
Set/Enable Watchdog
1004
Decrement Watchdog
1005
Watchdog Suspended?
1006
YES
Continue Boot
1008
NO
Watchdog Expired?
1010
YES
Trigger Watchdog Event
1012
End
NO
Boot reached specified point?
1014
YES
Boot is Successful
1016
End
NO

FIGURE 11

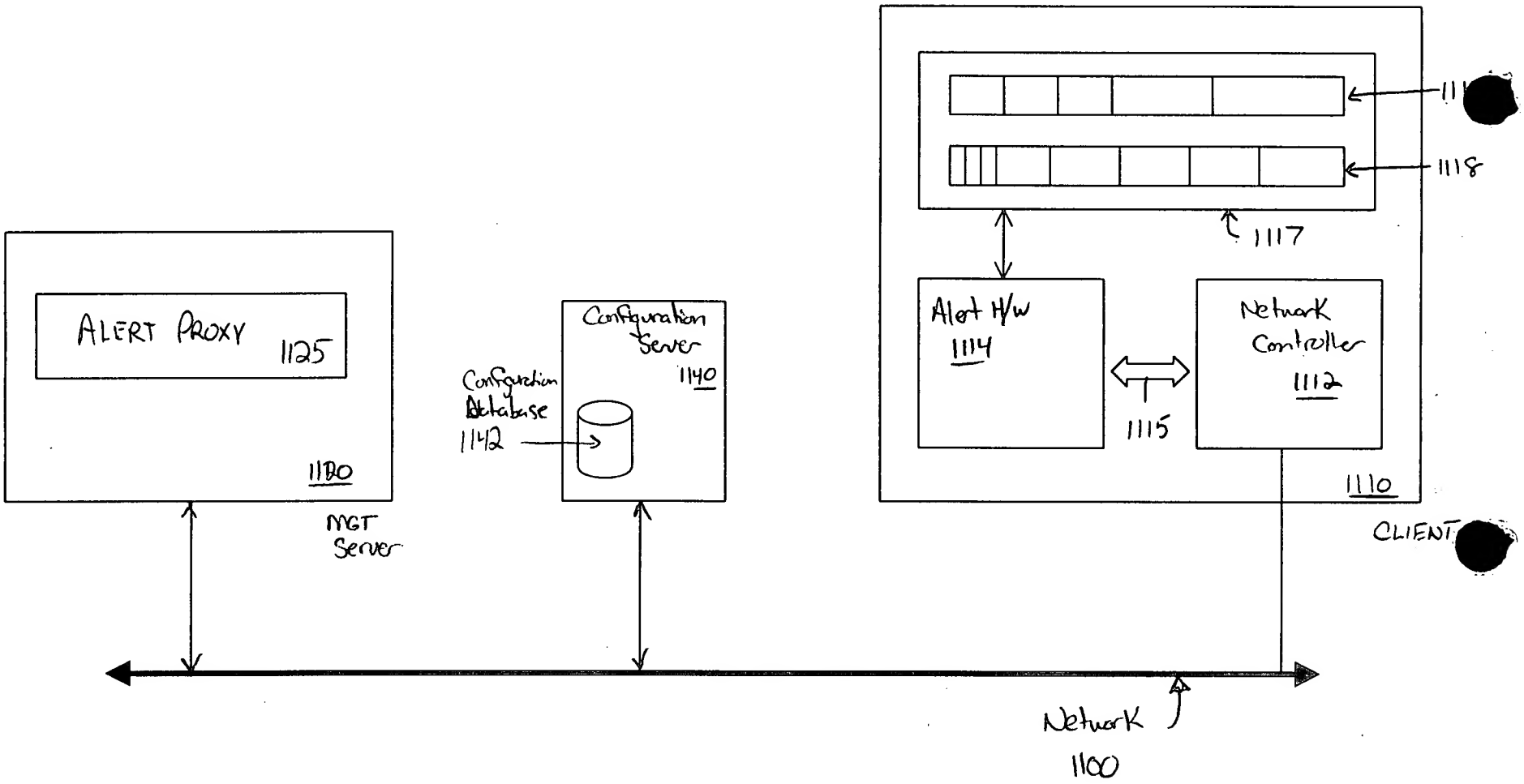
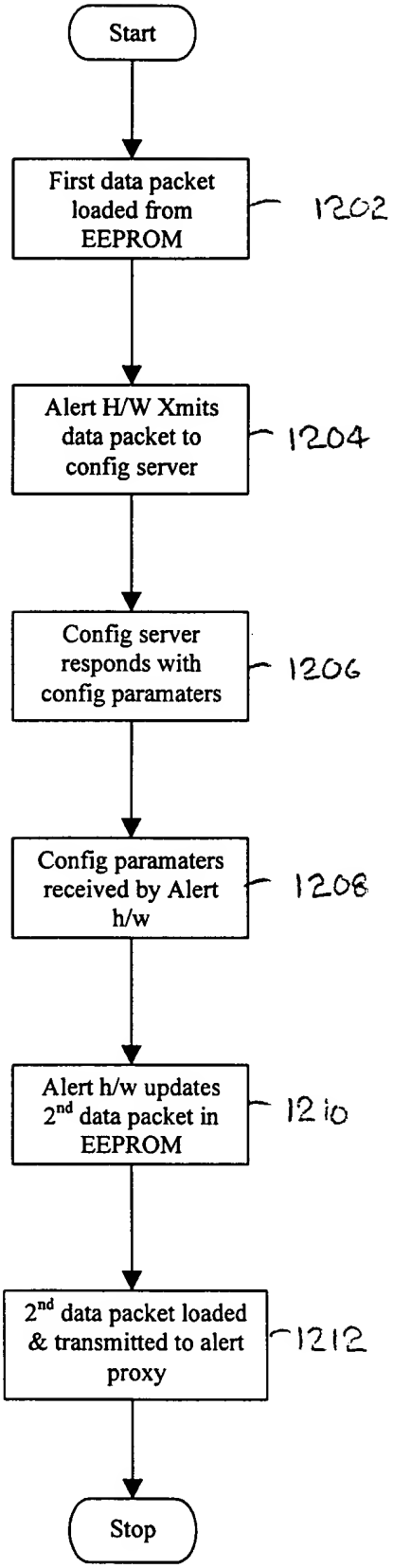


FIGURE 12


Start
First data packet loaded from EEPROM
1202
Alert H/W Xmits data packet to config server
1204
Config server responds with config paramaters
1206
Config paramaters received by Alert h/w
1208
Alert h/w updates 2nd data packet in EEPROM
1210
2nd data packet loaded & transmitted to alert proxy
1212
Stop